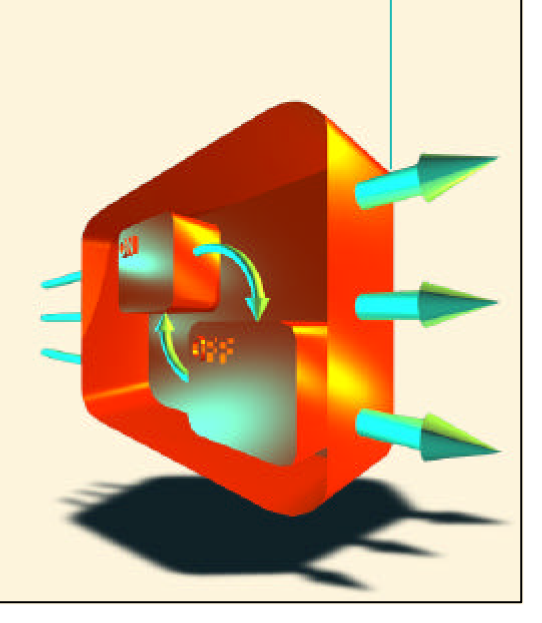
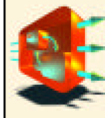
建立基本的StateFlow模型



元智大學
機械系
陳傳生 博士



課程目標

- 了解Simulink/Stateflow環境內的術語.
- 使用Graphic Edition建立stateflow diagram.
- 建立包含Stateflow block的基本 Simulink 模型.
- 建立Stateflow/Simulink之間的資料交換介面.
- 模擬Stateflow/Simulink的執行.

Copyright © 1984 - 1999 by The MathWorks, Inc.

How Does Stateflow Work?

建立及模擬一個Simulink/Stateflow的模型將用到下列工具

Graphics Editor

Dialog Boxes

Debugger

Data Dictionary

Explorer

Simulink

Finder

Generated Code

Copyright © 1984 - 1999 by The MathWorks, Inc.

術語Terminology

Stateflow Machine

The collection of all Stateflow diagrams contained within a single Simulink model.

Stateflow Block

A block inside a Simulink diagram that represents Stateflow.

Stateflow Diagram or Chart

The Stateflow contents that are contained in a single graphical window and correspond to one Stateflow Block.

one-to-one

Copyright © 1984 - 1999 by The MathWorks, Inc.

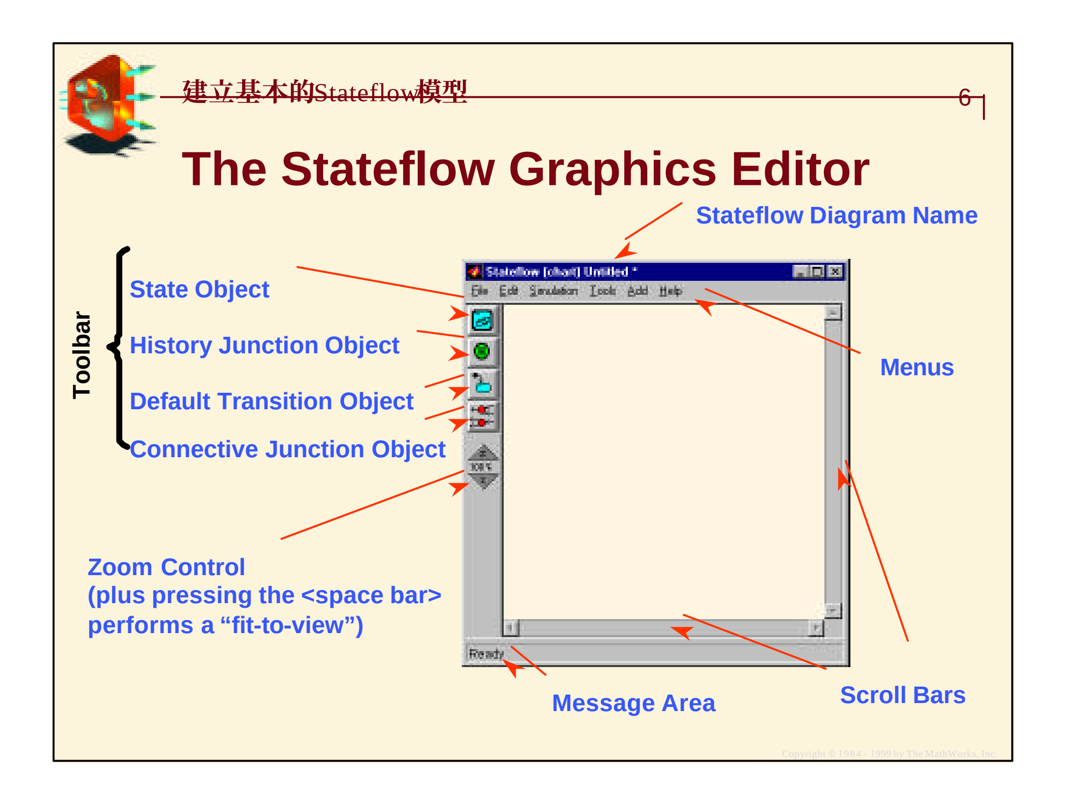
Creating a New Simulink Model and Stateflow Diagram

- Type **sfnew** to open a new Simulink model containing a Stateflow block
- Alternatively, the Stateflow block may be found in the Stateflow Blockset of Simulink

Double click on the Stateflow block to open the **graphics editor**

» sfnew

Copyright © 1984 - 1999 by The MathWorks, Inc.



Graphics Editor Menus

Stateflow [chart] Untitled *
File Edit Simulation Tools Add Help

Help: Open the online help (HTML)

Add: Create a new data or event object

Tools: Open other Stateflow tools: Explorer, Finder, Debugger, Simulation Target

Simulation: Open the Simulation Parameter dialog box, start a simulation

Edit: Copy, paste, and change the format (font, color, size, etc…) of selected objects.

File: Save or print current model; open a new model

Copyright © 1984 - 1999 by The MathWorks, Inc.

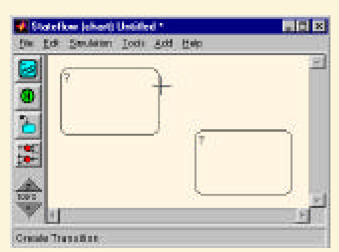
Adding objects from the Toolbar

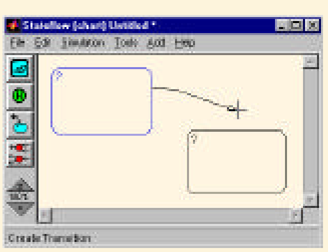
Method 1: Drag a single copy
(left click on the toolbar object, then left click on the graphics editor palette)

Method 2: Making multiple copies
(Double left click on the toolbar object; left click to place each copy; right click or *Esc* to stop)

Copyright © 1984 - 1999 by The MathWorks, Inc.

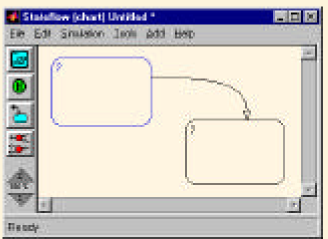
Creating Transitions





1. Position cursor near source object so that the cursor becomes a crosshair.

2. Drag the mouse to create the transition.

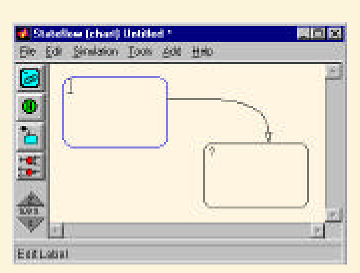


3. Release the mouse button at the destination object.

Copyright © 1984 - 1999 by The MathWorks, Inc.

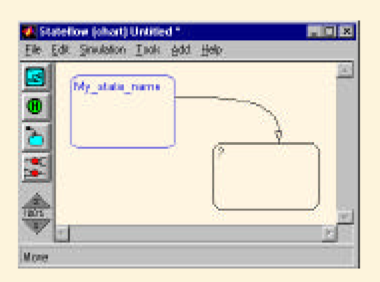


Labeling Objects



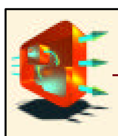
1. Place the cursor over the ? mark and left click once.
(Transitions must be selected first by left clicking on them.)

2. When the blinking cursor appears, type the label. Labels can be several lines long, 不能有空格

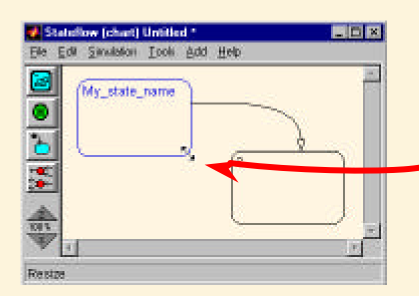


Labels are edited in the same way as they are created.

Copyright © 1984 - 1999 by The MathWorks, Inc.



Resizing States

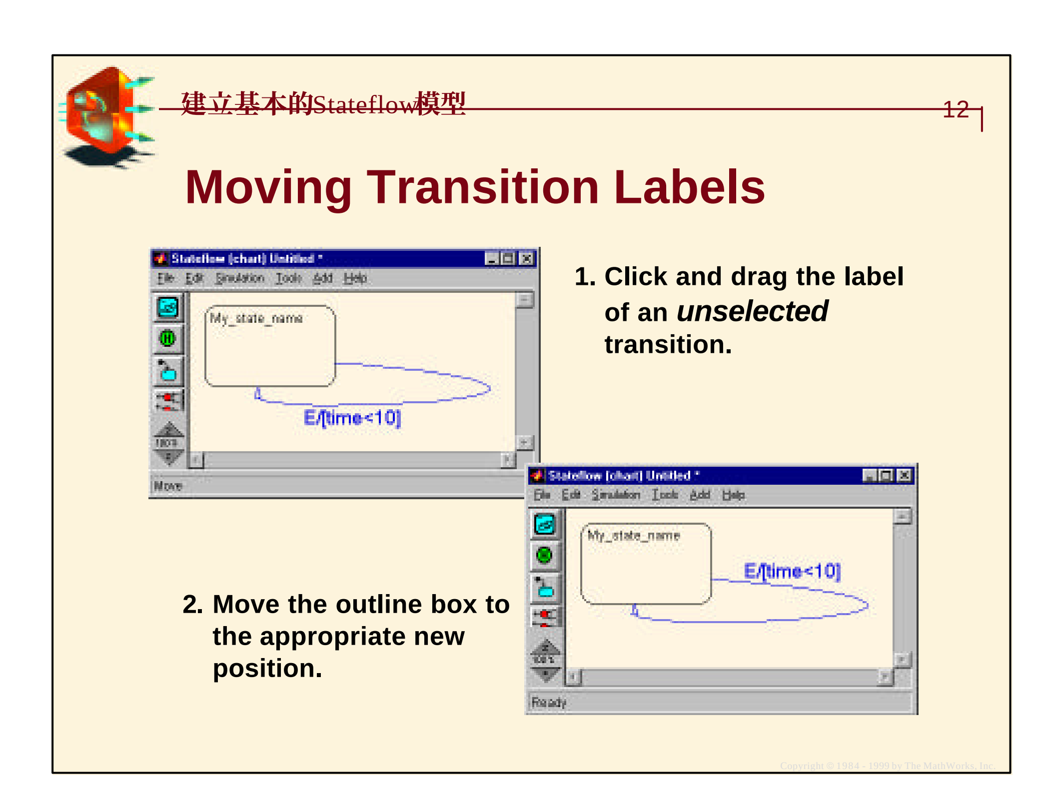


1. Place cursor over a corner of a state; the cursor will become a bi-directional arrow.

2. Press left mouse button, then drag to appropriately resize the state.

3. Release the mouse button

Copyright © 1984 - 1999 by The MathWorks, Inc.



Moving Transition Labels

1. Click and drag the label of an ***unselected*** transition.

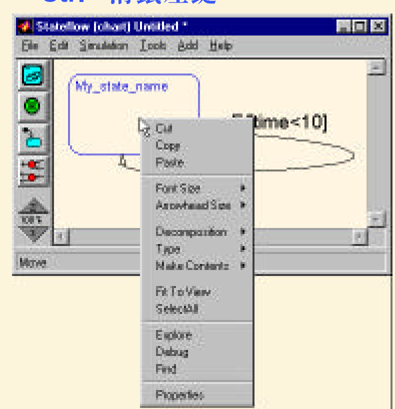
2. Move the outline box to the appropriate new position.

Copyright © 1984 - 1999 by The MathWorks, Inc.

Shortcut Menu

- Used to customize the Stateflow diagram appearance.
- Each graphical object and the overall Stateflow diagram (chart) has one.
- Can be applied to multiple objects.

滑鼠右鍵
或是
Ctrl+ 滑鼠左鍵



Copyright © 1984 - 1999 by The MathWorks, Inc.

Shortcut Menu

Editing: Cut, Copy, Paste

調整大小: Font Size, Arrowhead Size

Parallel or Exclusive: Decomposition

State or Box: Type

Grouped/Ungrouped and Shown/Hidden: Make Contents

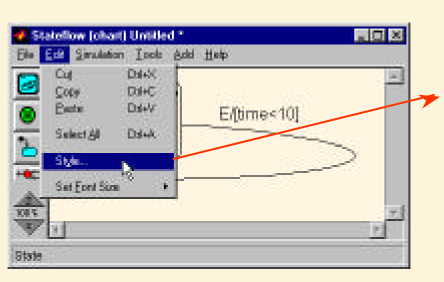
Fit To View
SelectAll

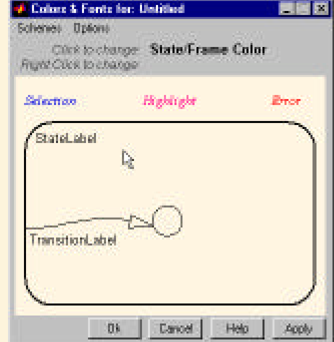
啟動其他工具: Explore, Debug, Find

Open the objects dialog box: Properties

Copyright © 1984 - 1999 by The MathWorks, Inc.

改變顏色



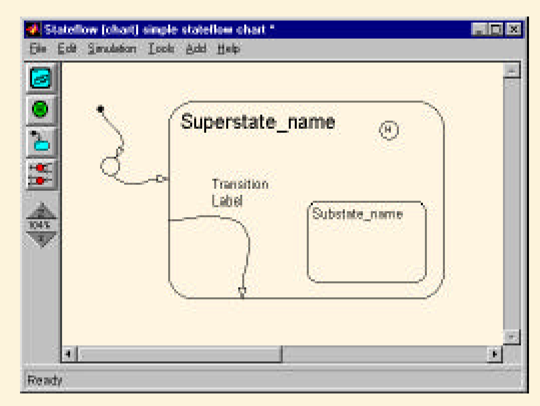


From the *Edit/Style* pulldown menu, the colors of states, transitions, labels, etc can be customized.

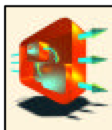
Copyright © 1984 - 1999 by The MathWorks, Inc.

練習 1: Using the Editor

Create this Stateflow diagram:

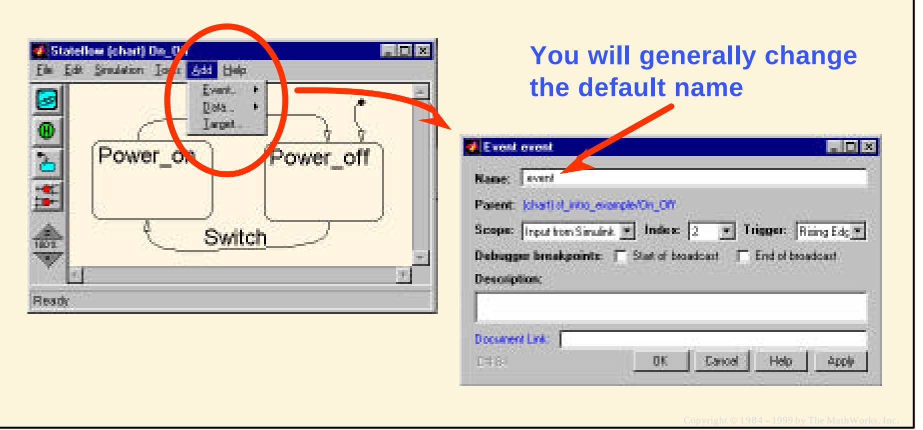


Copyright © 1984 - 1999 by The MathWorks, Inc.



Defining Data and Events

- You must define data and events before a Stateflow diagram can be properly parsed.
- Use the Add menu to define a new data or event object; this will invoke the appropriate dialog.



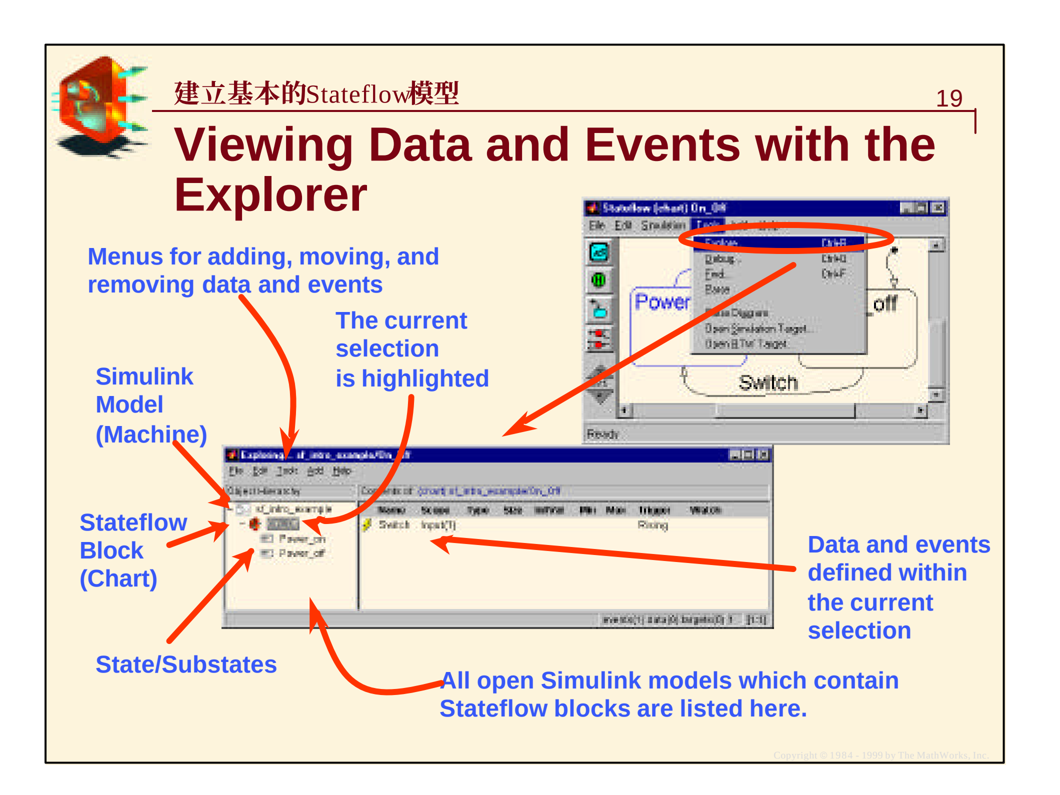
Copyright © 1984 - 1999 by The MathWorks, Inc.

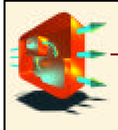
Properties Menu

- Fields will vary depending on the object selected (e.g. chart, state, transition, etc…)
- Property dialog boxes allow you to modify non-graphical information about objects.
- All user-created objects have property dialog boxes.



Copyright © 1984 - 1999 by The MathWorks, Inc.





Editing Data and Events with the Explorer

- To edit properties of defined data or events with the Explorer, use the following mouse clicks:
  - Left click on data or events to see possible settings and change them (including the name).
  - Double left click on data or events to open its properties menu.
  - Right click to see a short-cut menu.

Note:可以同時選取及修改多個data 或events

- Data and events may be moved within the model by using simple *drag and drop* operations.

Copyright © 1984 - 1999 by The MathWorks, Inc.



Stateflow/Simulink Interfacing

Simulink透過Data 及 Events 與Stateflow溝通

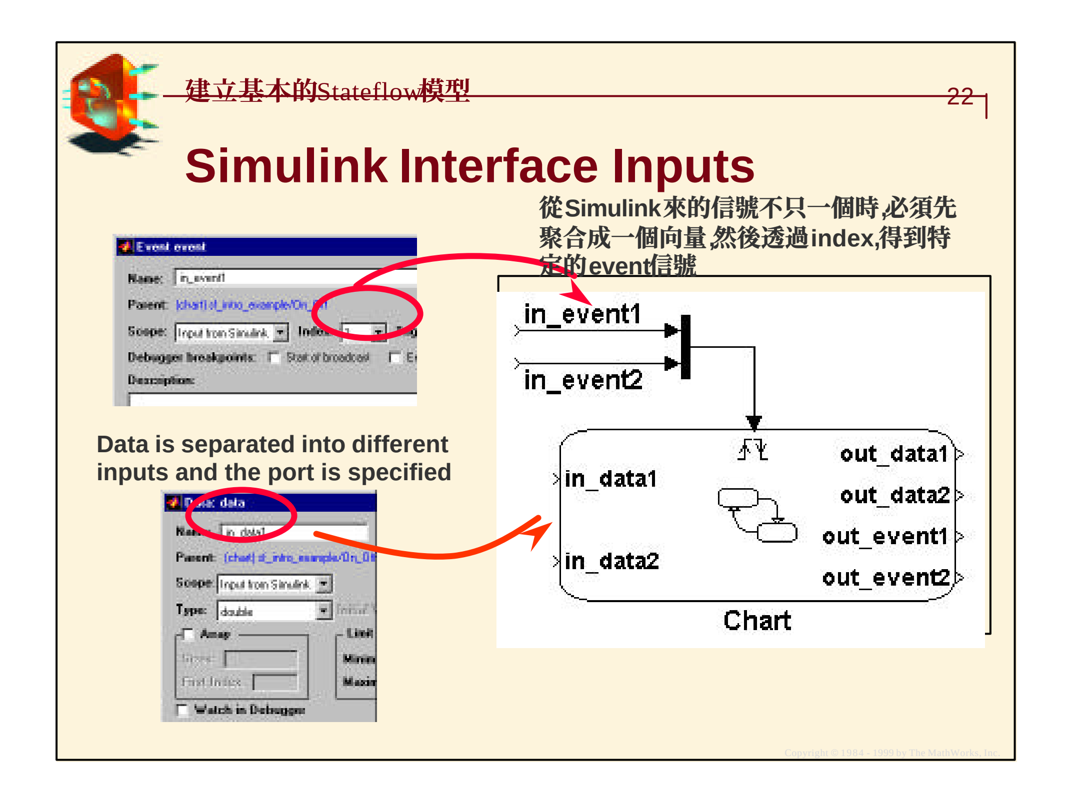
輸入

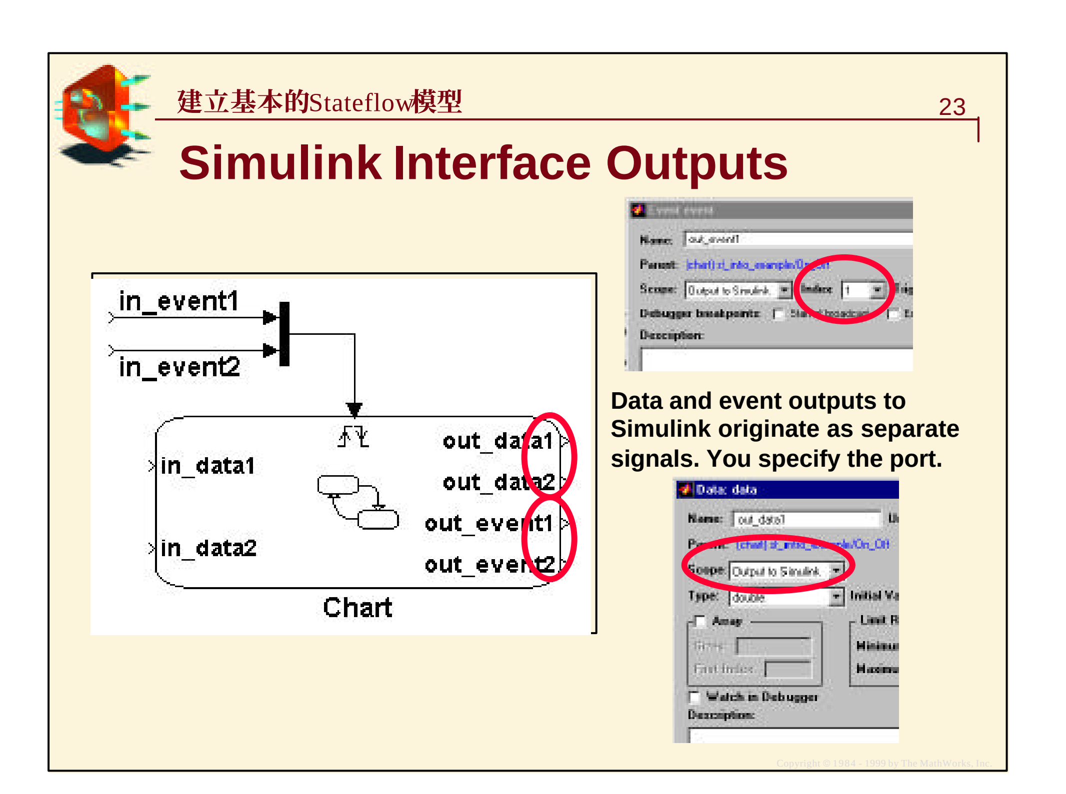
- Data inputs represent numerical values, the usual signal flow within Simulink.
- Simulink中的trigger 對應到 Stateflow 內的Event. 模擬時的triggers送入Stateflow chart時, 等於發生一個事件(Event).

輸出

- Data outputs are used as any other Simulink signal.
- 從Stateflow diagram送出的Event outputs可用來觸發其他Stateflow blocks或Simulink中的triggered subsystems.

Copyright © 1984 - 1999 by The MathWorks, Inc.





What does Event Driven Mean?

The Stateflow chart only executes when its triggering signal crosses zero.

Suppose this is the continuous signal triggering a Stateflow block

y

t

1/s

y

Integrator

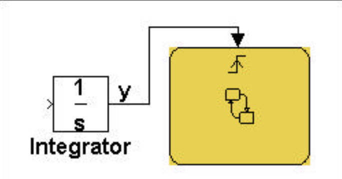
Copyright © 1984 - 1999 by The MathWorks, Inc.

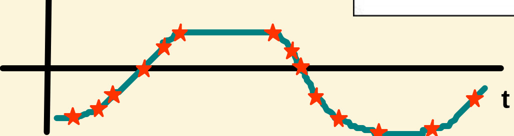


What does Event Driven Mean?

The Stateflow chart only executes when its triggering signal crosses zero.

The Simulink solver determines the required time points for simulation.



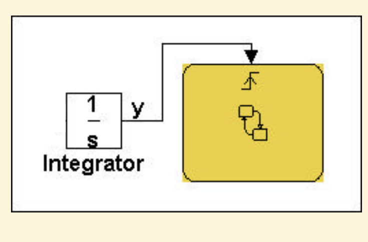


Copyright © 1984 - 1999 by The MathWorks, Inc.

What does Event Driven Mean?

The Stateflow chart only executes when its triggering signal crosses zero.

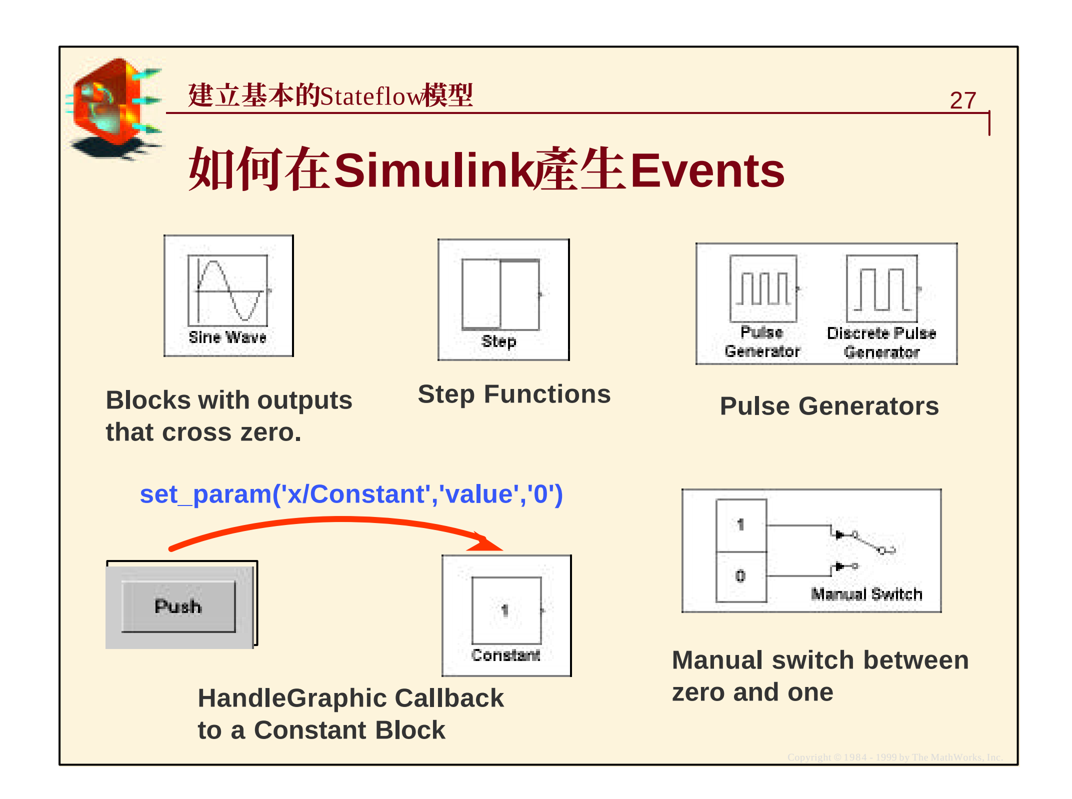
At times of zero crossing the Stateflow block executes.



t

The Stateflow block executes as a *result* of another signal.

Copyright © 1984 - 1999 by The MathWorks, Inc.



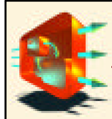
Demo Model

Create a model that determines whether there has been an even or odd number of zero-crossings.

Set Max step size to 1e-4

» oddeven

Copyright © 1984 - 1999 by The MathWorks, Inc.



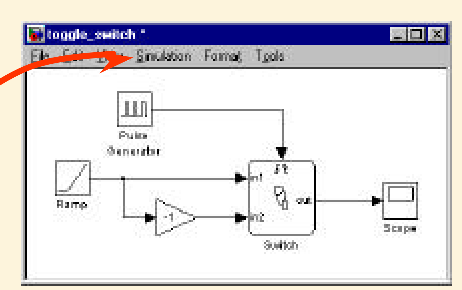
練習 2: Toggle Switch

建立一項模擬toggle switch的StateFlow 模型:

- 使用兩個states及兩個transitions在兩個states間轉換.
- 當輸入的切換信號改變時,它的狀態改變成另一個狀態.
- 包含兩個data inputs及一個data output. 輸出值在兩個輸入值切換.

建立如右圖的模型,並且完成模擬:

- 模擬時間 40 秒
- 使用一個週期4秒的 pulse generator

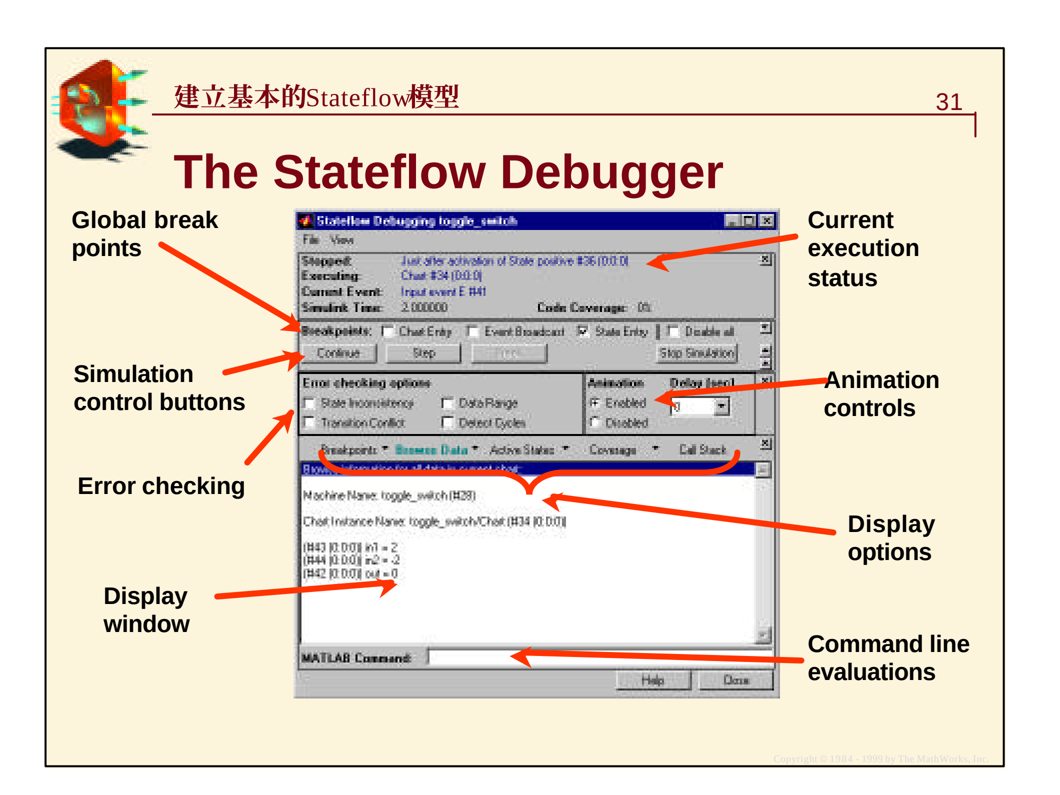


Copyright © 1984 - 1999 by The MathWorks, Inc.

Solution: Toggle Switch

» toggle_switch

Copyright © 1984 - 1999 by The MathWorks, Inc.



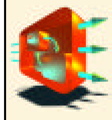
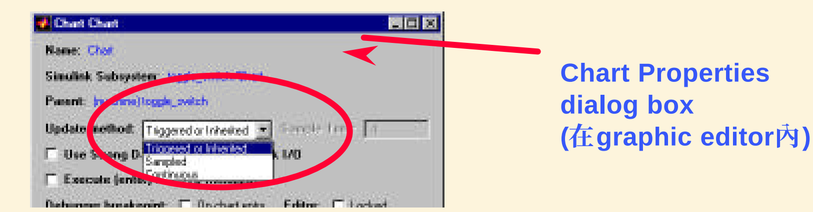


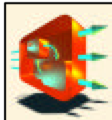
Chart Update Methods

有3種更新方式:

1. Triggered or inherited –The Stateflow block is triggered by input events or updates as its inputs change.
2. Sampled – 以固定速率更動Stateflow block.
3. Continuous – 每次Simulink執行integration step時,同時評估Stateflow block.

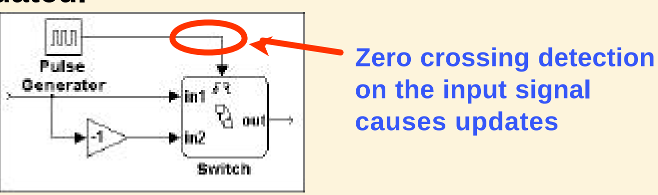


Copyright © 1984 - 1999 by The MathWorks, Inc.



Triggered / Inherited Updates

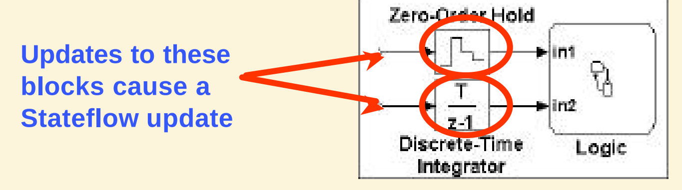
- Input triggers cause the Stateflow diagram to be evaluated.



Zero crossing detection on the input signal causes updates

- When no input events exist, the block inherits its updates from the blocks that pass in signals.

Updates to these blocks cause a Stateflow update



Copyright © 1984 - 1999 by The MathWorks, Inc.

Sampled Updates

The Stateflow block acts as a discrete block updating with a uniform sampling rate independent of the blocks that pass in signals.

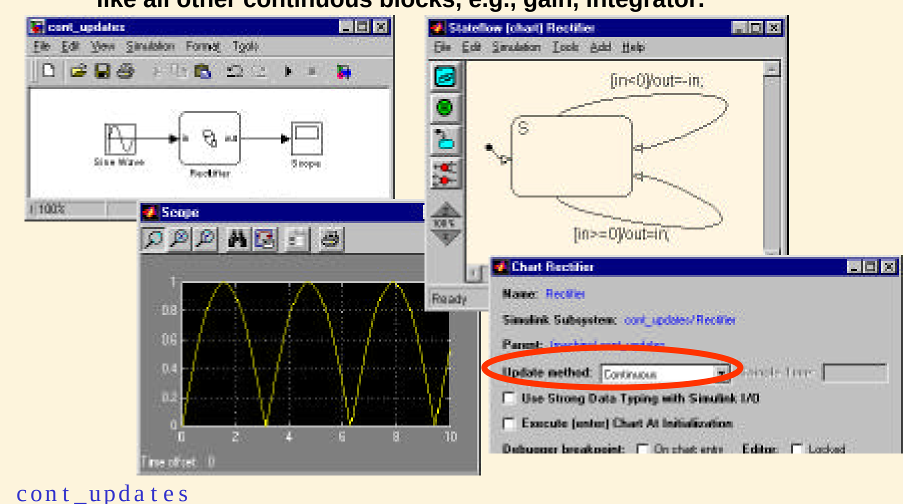
Update rate

» sampled_updates

Copyright © 1984 - 1999 by The MathWorks, Inc.

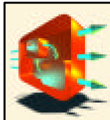
Continuous Updates

The Stateflow block is evaluated at each Simulation integration step, like all other continuous blocks, e.g., gain, integrator.



» cont_updates

Copyright © 1984 - 1999 by The MathWorks, Inc.



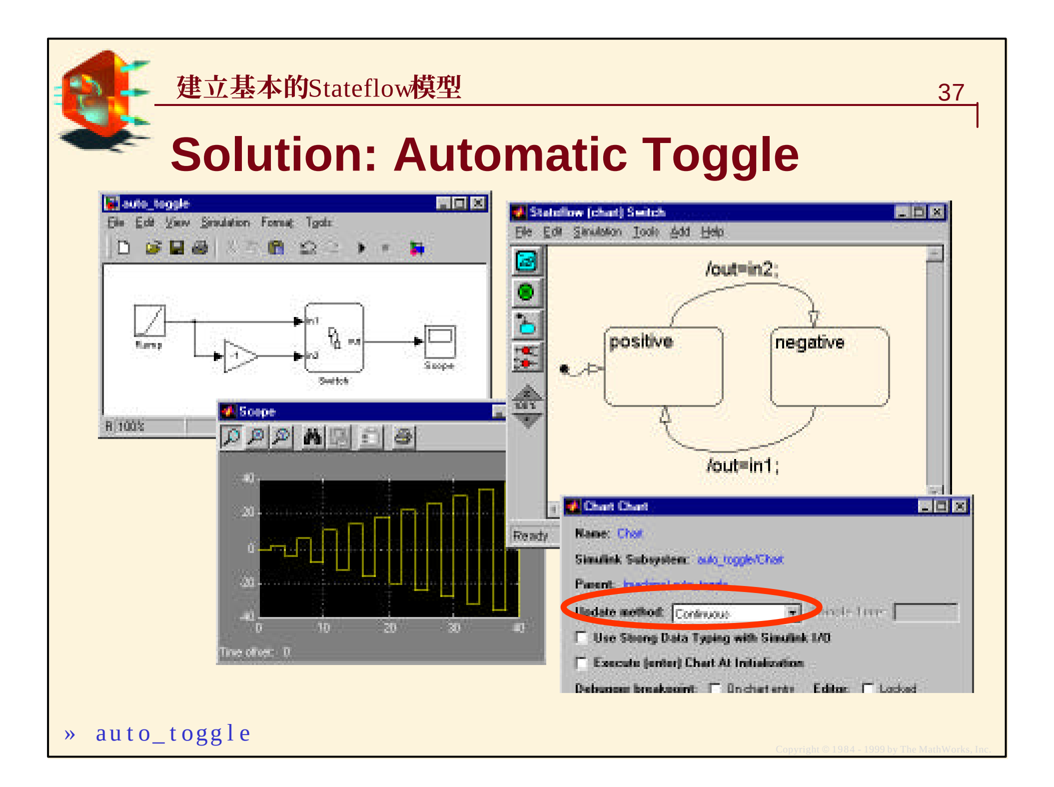
Exercise 3: Automatic Toggle

Modify your previous example to toggle automatically (without a triggering event) and verify that your output resembles your previous outputs.

Hint: Change the chart update method and delete the input event using the Explorer.

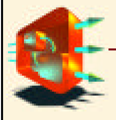
Bonus: Explain why the output using this update method is different from that obtained using explicit event triggering.

Copyright © 1984 - 1999 by The MathWorks, Inc.



» auto_toggle

Copyright © 1984 - 1999 by The MathWorks, Inc.



Summary

- Working Stateflow Blocks are built by drawing graphical objects with the graphics editor and defining non-graphical objects with menus.
- Stateflow Blocks are event driven which implies that a triggering signal must cross zero before execution will occur. By selecting continuous or sampled update methods implicit events are generated from Simulink to execute the Stateflow Diagrams.
- Stateflow Diagrams are built with a number of different tools all of which communicate information to the data dictionary.

Copyright © 1984 - 1999 by The MathWorks, Inc.